



IN THE CLAIMS:

1. (Currently Amended) A method for programmatically generating a second graphical program based on a first graphical program, the method comprising:

receiving information specifying the first graphical program, wherein the first graphical program includes a first plurality of interconnected nodes which perform a first functionality, and wherein the first graphical program is associated with a first programming development environment; and

programmatically generating the second graphical program based on the information, wherein the second graphical program implements functionality of the first graphical program, wherein the second graphical program includes a second plurality of interconnected nodes which perform the first functionality, wherein the second graphical program is associated with a second programming development environment, and wherein the second programming development environment is different from the first programming development environment.

2. (Original) The method of claim 1, wherein the second graphical program performs substantially like the first graphical program.

RECEIVED

MAY 11 2004

Technology Center 2100

3. (Original) The method of claim 1,
wherein the first graphical program implements first functionality;
wherein the second graphical program also implements the first functionality.

4. (Original) The method of claim 1,
wherein the second graphical program implements only a portion of functionality of the first graphical program.

5. (Cancelled)

6. (Currently Amended) The method of claim [[5]] 1, wherein the first plurality of interconnected nodes in the first graphical program visually indicate the first functionality of the first graphical program;

wherein the second plurality of interconnected nodes in the second graphical program visually indicate the first functionality of the second graphical program.

7. (Currently Amended) The method of claim [[5]]1,
wherein the first plurality of interconnected nodes are interconnected according to one or more of data flow, control flow, and execution flow;
wherein the second plurality of interconnected nodes are interconnected according to one or more of data flow, control flow, and execution flow.

8. (Original) The method of claim 1,
wherein the first graphical program includes a first block diagram and a first user interface;
wherein the second graphical program includes a second block diagram and a second user interface.

9. (Original) The method of claim 8,
wherein the second block diagram appears substantially like the first block diagram.

10. (Original) The method of claim 8,
wherein the second user interface appears substantially like the first user interface.

11. (Cancelled)

12. (Original) The method of claim 1,
wherein the first graphical program is developed according to a first programming development environment;
wherein the second graphical program is programmatically generated according to a second programming development environment, wherein the second programming development environment is different than the first programming development environment.

13. (Original) The method of claim 1,
wherein the first graphical program is developed according to a first graphical programming language;

wherein the second graphical program is programmatically generated according to a second graphical programming language, wherein the second graphical programming language is different than the first graphical programming language.

14. (Original) The method of claim 13,
wherein the first graphical programming language is the G language.

15. (Original) The method of claim 13,
wherein the second graphical programming language is the G language.

16. (Original) The method of claim 1,
wherein the first graphical program is one of a data flow program, control flow program, or an execution flow program;
wherein the second graphical program is one of a data flow program, control flow program, or an execution flow program.

17. (Original) The method of claim 1, further comprising:
executing the second graphical program, wherein the second graphical program performs functionality of the first graphical program.

18. (Original) The method of claim 1, further comprising:
compiling the second graphical program;
downloading the second graphical program to a hardware device.

19. (Original) The method of claim 1,
wherein the first graphical program is a Simulink program. (Original)

20. (Original) The method of claim 1,

wherein the first graphical program is one of a Simulink program, a VEE program, or a LabVIEW program.

21. (Original) The method of claim 1,
wherein the second graphical program is one of a Simulink program, a VEE program, or a LabVIEW program.

22. (Original) The method of claim 1,
wherein the received information specifying the first graphical program comprises one or more of text information and binary information.

23. (Original) The method of claim 1,
wherein the received information specifying the first graphical program comprises abstract information describing the first graphical program.

24. (Original) The method of claim 1,
wherein the received information specifying the first graphical program comprises a model file describing the first graphical program.

25. (Previously Presented) The method of claim 1, further comprising:
constructing an abstract representation of the first graphical program based on the received information specifying the first graphical program;
wherein said programmatically generating the second graphical program comprises programmatically creating the second graphical program based on the constructed representation.

26. (Original) The method of claim 25,
wherein said constructing an abstract representation of the first graphical program comprises constructing a directed graph representation of the first graphical program.

27. (Original) The method of claim 25,

wherein said constructing an abstract representation of the first graphical program comprises constructing one or more data structures representing the first graphical program.

28. (Original) The method of claim 1,
wherein the first graphical program includes a first node;
wherein said receiving information specifying the first graphical program comprises receiving information specifying the first node;
wherein said programmatically generating the second graphical program comprises programmatically including a second node corresponding to the first node in the second graphical program.

29. (Original) The method of claim 28,
wherein the first node has particular functionality;
wherein the second node has functionality equivalent to the particular functionality of the first node.

30. (Currently Amended) A memory medium for programmatically generating a second graphical program based on a first graphical program, the memory medium comprising program instructions executable to:

receive information specifying the first graphical program, wherein the first graphical program includes a first plurality of interconnected nodes which perform a first functionality, and wherein the first graphical program is associated with a first programming development environment; and

programmatically generate the second graphical program based on the information, wherein the second graphical program implements functionality of the first graphical program, wherein the second graphical program includes a second plurality of interconnected nodes which perform the first functionality, wherein the second graphical program is associated with a second programming development environment, and wherein the second programming development environment is different from the first programming development environment.

31. (Original) The memory medium of claim 30, wherein the second graphical program performs substantially like the first graphical program.

32. (Original) The memory medium of claim 30,
wherein the first graphical program implements first functionality;
wherein the second graphical program also implements the first functionality.

33. (Original) The memory medium of claim 30,
wherein the second graphical program implements only a portion of functionality
of the first graphical program.

34. (Cancelled)

35. (Currently Amended) The memory medium of claim ~~[[34]]~~30,
wherein the first plurality of interconnected nodes are interconnected according to
one or more of data flow, control flow, and execution flow;
wherein the second plurality of interconnected nodes are interconnected according
to one or more of data flow, control flow, and execution flow.

36. (Original) The memory medium of claim 30,
wherein the first graphical program includes a first block diagram and a first user
interface;
wherein the second graphical program includes a second block diagram and a
second user interface.

37. (Cancelled)

38. (Currently Amended) A system for programmatically generating a second
graphical program based on a first graphical program, the system comprising:
a processor coupled to a memory, wherein the memory stores program
instructions;

wherein the processor is operable to execute the program instructions in order to:
receive information specifying the first graphical program, wherein the first graphical program includes a first plurality of interconnected nodes which perform a first functionality, and wherein the first graphical program is associated with a first programming development environment; and
programmatically generate the second graphical program based on the information, wherein the second graphical program implements functionality of the first graphical program, wherein the second graphical program includes a second plurality of interconnected nodes which perform the first functionality, wherein the second graphical program is associated with a second programming development environment, and wherein the second programming development environment is different from the first programming development environment.

39. (Original) The system of claim 38,
wherein the first graphical program implements first functionality;
wherein the second graphical program also implements the first functionality.

40. (Original) The system of claim 38,
wherein the second graphical program implements only a portion of functionality of the first graphical program.

41. (Cancelled)

42. (Original) The system of claim 38,
wherein the first plurality of interconnected nodes are interconnected according to one or more of data flow, control flow, and execution flow;
wherein the second plurality of interconnected nodes are interconnected according to one or more of data flow, control flow, and execution flow.

43. (Original) The system of claim 38,
wherein the first graphical program includes a first block diagram and a first user interface;

wherein the second graphical program includes a second block diagram and a second user interface.

44. (Cancelled)

45. (New) The method of claim 1, wherein the information specifying the first graphical program comprises the first graphical program.

46. (New) The method of claim 1, wherein the first graphical program is a first type of data flow diagram;

wherein the second graphical program is a second type of data flow diagram.

47. (New) The method of claim 1, wherein the first graphical program comprises one or more loops among block diagram node interconnections;

wherein the second graphical program does not comprises one or more loops among block diagram node interconnections.

48. (New) A carrier medium for programmatically generating a second graphical program based on a first graphical program, the memory medium comprising program instructions executable to:

receive the first graphical program, wherein the first graphical program comprises a first plurality of interconnected nodes which perform a first functionality, and wherein the first graphical program is associated with a first programming development environment; and

programmatically generate the second graphical program based on the first graphical program, wherein the second graphical program implements functionality of the first graphical program, wherein the second graphical program comprises a second plurality of interconnected nodes which perform the first functionality, wherein the second graphical program is associated with a second programming development environment, and wherein the second programming development environment is different from the first programming development environment.

49. (New) A carrier medium for programmatically generating a second graphical program based on a first graphical program, the memory medium comprising program instructions executable to:

receive the first graphical program, wherein the first graphical program comprises a first plurality of interconnected nodes which perform a first functionality, and wherein the first graphical program is in a first graphical programming language; and

programmatically generate the second graphical program based on the first graphical program, wherein the second graphical program implements functionality of the first graphical program, wherein the second graphical program comprises a second plurality of interconnected nodes which perform the first functionality, wherein the second graphical program is in a second graphical programming language, and wherein the second graphical programming language is different from the first graphical programming language.

50. (New) The carrier medium of claim 49,

wherein the first graphical program includes a first block diagram and a first user interface; and

wherein the second graphical program includes a second block diagram and a second user interface.

51. (New) The carrier medium of claim 50,

wherein the second block diagram appears substantially like the first block diagram.

52. (New) The carrier medium of claim 50,

wherein the second user interface appears substantially like the first user interface.

53. (New) A carrier medium for programmatically generating a second graphical program based on a first graphical program, the memory medium comprising program instructions executable to:

receive the first graphical program, wherein the first graphical program comprises a first plurality of interconnected nodes which perform a first functionality, and wherein the first graphical program has first data flow semantics; and

programmatically generate the second graphical program based on the first graphical program, wherein the second graphical program implements functionality of the first graphical program, wherein the second graphical program comprises a second plurality of interconnected nodes which perform the first functionality, wherein the second graphical program has second data flow semantics, and wherein the second data flow semantics is different from the first data flow semantics.

54. (New) The carrier medium of claim 53,

wherein the first graphical program includes a first block diagram and a first user interface; and

wherein the second graphical program includes a second block diagram and a second user interface.

55. (New) The carrier medium of claim 54,

wherein the second block diagram appears substantially like the first block diagram.

56. (New) The carrier medium of claim 54,

wherein the second user interface appears substantially like the first user interface.